AFRL-IF-RS-TR-2001-76 Vol I (of four)
Final Technical Report
May 2001

EXPLORING A THEORY DESCRIBING THE PHYSICS OF INFORMATION SYSTEMS, A PHYSICAL MODEL OF THE BEHAVIOR OF INFORMATION SYSTEMS

Zetetix

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. K177

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2001-76 Vol. 1 (of four) has been reviewed and is approved for publication.

APPROVED: Deborah A. Cerino

DEBORAH A. CERINO
Project Engineer

FOR THE DIRECTOR: James A. Collins

JAMES A. COLLINS, Acting Chief
Information Technology Division
Information Directorate

If your address has changed or if you wish to be removed from the Air Force Research Laboratory Rome Research Site mailing list, or if the addressee is no longer employed by your organization, please notify AFRL/IFTD, 525 Brooks Road, Rome, NY 13441-4505. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

EXPLORING A THEORY DESCRIBING THE PHYSICS OF INFORMATION SYSTEMS, A PHYSICAL MODEL OF THE BEHAVIOR OF INFORMATION SYSTEMS

Scott Young Harmon

Contractor:	Zetetix
Contract Number:	F30602-00-C-0104
Effective Date of Contract:	06 April 2000
Contract Expiration Date:	05 October 2000
Short Title of Work:	Exploring a Theory Describing the Physics of Information Systems, A Physical Model of the Behavior of Information Systems
Period of Work Covered:	Apr 00 - Oct 00
Principal Investigator:	Scott Young Harmon
Phone:	(818) 991-0480
AFRL Project Engineer:	Deborah A. Cerino
Phone:	(315) 330-1445

This research was supported by the Defense Advanced Research Projects Agency of the Department of Defense and was monitored by Deborah A. Cerino, AFRL/IFTD, 525 Brooks Road, Rome, NY.

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	MAY 2001	Final Apr 00 - Oct 00

4. TITLE AND SUBTITLE
EXPLORING A THEORY DESCRIBING THE PHYSICS OF INFORMATION SYSTEMS, A PHYSICAL MODEL OF THE BEHAVIOR OF INFORMATION SYSTEMS

5. FUNDING NUMBERS
C - F30602-00-C-0104
PE - 63760E
PR - IAST
TA - 00
WU - P1

6. AUTHOR(S)
Scott Young Harmon

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Zetetix
PO Box 2640
Agoura CA 91376-2640

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Defense Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington VA 22203-1719

Air Force Research Laboratory/IFTD
525 Brooks Road
Rome New York 13441-4505

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2001-76
Vol 1 (of four)

11. SUPPLEMENTARY NOTES
Air Force Research Laboratory Project Engineer: Deborah A. Cerino/IFTD/(315) 330-1445

12a. DISTRIBUTION AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
This project accomplished all of its objectives: document a theory of information physics, conduct a workshop on planning experiments to test this theory, and design experiments that validate this theory. Information physics proposes quantitative relationships between observable information flows and changes in the content information systems maintain. This theory explains all flows within information systems as either diffusive or force-driven. The forces driving information flows arise from the existence of goal content. The workshop participants discussed various theories and considered experiments that characterize the macroscopic phenomena underlying complex information system behavior. These participants identified experimental opportunities that exploit existing databases, execute simulations and conduct traditional controlled experiments. They recommended that focussed experiments to test theories explaining information system phenomena were feasible today. The experiment plan builds upon the workshop's result and proposes experiments that measure information device energy dissipation, test the independence of symbol execution work from device efficiency, measure information diffusion rates in information systems, and measure force-driven information flows. These experiments are both technically and programmatically feasible. When validated, the proposed theory can guide designers to reliably build more effective, secure and predictable information systems.

14. SUBJECT TERMS
Information Physics, Physics of Computation, Information Theory, Thermodynamics

15. NUMBER OF PAGES
60

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

TABLE OF CONTENTS

List of Figures

List of Tables

Abstract

The model of information systems, discussed in this paper, builds upon past work in the physics of computation, computational complexity theory, and information theory. This model represents the impact that existing physical laws have upon the state and behavior of objects in the abstract worlds that information systems create, maintain, store and communicate. It posits that information exists only as modulated energy quantized into abstract symbols. Two forms of these symbols exist, pure data and instructions. The devices in information systems can execute instruction symbols. This execution process transforms pure data inputs into pure data outputs. All information flows through processing, communication links and memory require the devices of an information system to perform physical work. This property suggests that the work required to support an information flow is proportional to the rate of that flow times the resistance that the device exerts against the flow. This model also suggests that the existence of a particular type of state information, as goals, in a system can drive the information flows within that system. The rate of goal-driven information flow is proportional to the potential of the driving goal divided by the device resistance.

Another form of information flow is analogous to diffusion. The rate of information diffusion within a system is proportional to a diffusion constant times the gradient of information complexity. Information complexity is proportional to the number of executable dependencies that could exist between the different abstract property states represented within that system. The observability of both information flow rates and content complexity permits the calculation of diffusion constants for different information systems. The similarities between highly complex systems and disordered systems suggests that system complexity, and therefore information complexity, is proportional to the physical entropy of a system. Since information contributes to system complexity, it also lends components to system entropy. This linkage between the entropy of a system and the information that could influence that system's behavior permits extending the Second Law of Thermodynamics to explain such phenomena as information leakage and progressive resource saturation. These two phenomena are commonly observed in complex information systems of today.

Introduction

Recent large scale information system failures with enormous costs to such companies as e-Bay, MCI Worldcom and Charles Schwab only betray our poor understanding of the phenomena governing information system behavior. No one controlling those systems expected those failures. Similar failures have occurred in military command and control systems. Under war time circumstances, failures of this same nature could have disastrous consequences. Only fundamental knowledge about information system behavior can enable designers to avoid the pitfalls causing the failures we see today. Such knowledge can also support the prediction of information system behavior over which we may have little or no overt control. This will enable us to create information systems that reliably deliver their services under widely varying, and possibly unpredictable, quality of service demand loads. Despite this tremendous need, no coherent body of such knowledge exists. The lack of this knowledge will ultimately limit the complexity of the information systems that we can realized with today's technology.

This paper describes a preliminary physical model of some of the phenomena underlying the behavior of all information systems, regardless of their specific implementation or the conditions of their use. This model has grown from the extensive past work in information theory, computational complexity and the physics of computation. It begins by assuming, among other things, that the behavior of any information system can be completely explained by the properties of its devices and the character of the information it possesses. Like any model of a physical system, this model abstracts the real behavior of an information system in the hope of describing some portion of this behavior under some set of well defined conditions. It begins this abstraction process by associating information with modulated energy then identifying the classes of devices that manipulate and respond to those energy modulations. Energy modulations give way to symbols in the next level of abstraction and this step enables measurement of the information flows through devices and information stored within them. Symbols also exist in both executable and nonexecutable forms and those forms provide the means to abstract information content as object dependencies and properties, respectively. At the conceptual level, this model partitions abstract object properties into representations of actual state and desired state. This discussion describes several relationships between the state and behavior of an information system's devices and the state and behavior of the abstract objects represented by the information contained by that system. These relationships were derived in a manner analogous to that used when trying to understand traditional physical systems. In some cases, these relationships are only linear representations. But, this form of mathematical abstraction is commonly employed when trying to understand the behavior of complex physical systems.

Information System Model

The model, contained herein, essentially describes information system state and behavior as derived from our knowledge of the physical world. Figure 1 illustrates the major conceptual components of this model.

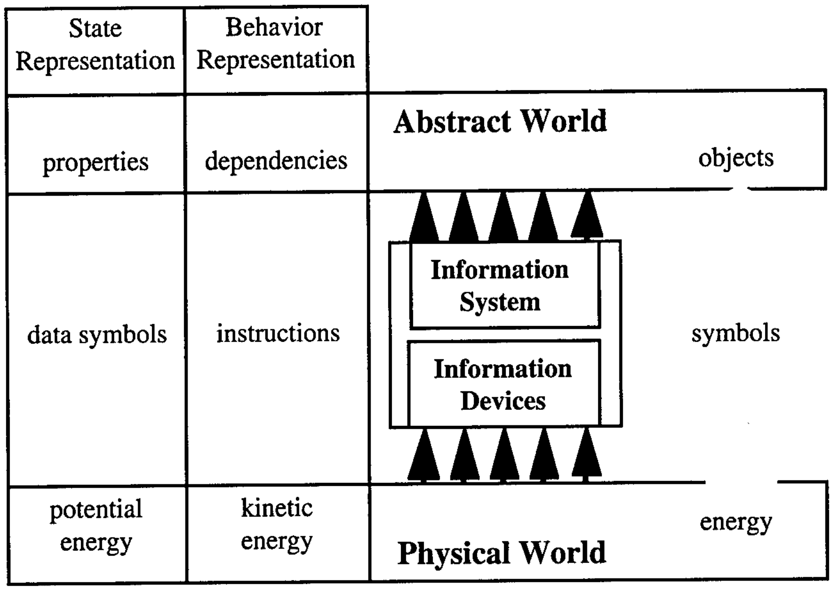


Figure 1. Model of Information System State and Behavior.

When information systems operate, they create, manipulate, store and communicate descriptions of an abstract world. Object properties and dependencies describe this abstract world's state and behavior. Information systems transform the symbols created and manipulated by information devices into the objects of their abstract world. Networks of interacting information devices compose all information systems. An information device can be a processor, router, memory writer, memory reader, communication transmitter or communication receiver. These devices couple the information system to the physical world by transforming undifferentiated energy into information symbols.

Development of the proposed information system model begins with the articulation of four key assumptions:

<u>Assumption 1:</u> An information system's externally observable behavior consists completely of the information it produces as output in response to the information it receives as input.

<u>Assumption 2:</u> The principles of thermodynamics constrain all of an information system's behavior (i.e., no magic, please).

<u>Assumption 3:</u> The properties of an information system's devices and the content of the information that system possesses completely determine that system's behavior under all conditions.

<u>Assumption 4:</u> Information flow and storage completely characterizes the state and behavior of all information systems.

Of course, the proposed model rests upon other assumptions but these will be presented where the development needs them.

Information as Modulated Energy

At the first level of abstraction, information exists simply as modulated energy. This permits identifying a few basic classes of information devices from which to construct all information systems and considers the flow of information at levels closest to their physical appearance.

Nature of Information

Few concepts possess such a broad range of meanings as the notion of information. Unfortunately, this notion forms the core that defines the functionality of all information systems and modeling this functionality comprehensively requires a clear and unambiguous definition of information. Definition 1 attempts to provide the clarity required by this paper while largely preserving the connection with the definitions encouraged by common usage.

<u>Definition 1:</u> Information - modulated energy, either kinetic or potential, whose modulations can change the state of at least one information system that receives it as input and that did not posses that information before receiving it.

Definition 1 inextricably couples the existence of information with the existence of energy and through that coupling captures the property that information is a physical quantity, as vigorously argued by Landauer [1-3] and Kantor [4]. The modulation of energy is commonly called encoding and has been treated richly by the discipline of information theory [5-7]. Spatially modulated potential energy enables storing information and temporally modulated kinetic energy enables communicating and processing information.

From Definition 1, information exists in a physical location if and only if four conditions are true:

- Energy exists in that location;
- That energy is modulated in some way;
- An information system also exists at that location; and
- The behavior of that information system can change as a result of possessing that information.

The latter conditions suggests a convenient and useful way to identify meaningful information and Definition 2 captures that way.

<u>Definition 2:</u> Meaningful information – that information that can potentially change the behavior of an information system receiving it that did not possess it beforehand.

All forms of energy, including matter, can host information. Generally, kinetic energy transports information through three dimensional space and potential energy stores information over time within a fixed space. However, exceptions to these generalizations do exist. For example, transporting magnetic media over a distance involves both forms of energy. Information can also be stored as kinetic energy as modulated light in a fiber optic ring. But, these exceptions violate none of the four conditions described above.

Similarly, any technique of energy modulation can encode information. The diversity produced by combining any modulation technique with any energy form creates the wealth of forms of information seen in the world today and hints at the vast possibilities that may exist in the future.

Clearly, the last two conditions demand some definition of an information system that avoids a possible circularity between the notions of information and information systems. A later section will present such a definition built upon the necessary foundation presented in the next section.

Information Devices

Assumption 5 correlates the existence of information with that of information devices.

Assumption 5: Information only exists within and because of the existence of information devices.

Information systems require information devices to create, manipulate, store, retrieve and communicate the information within them. In a sense, information devices form the layer that couples information to the physical energy that carries it by modulating that energy and responding to those modulations input to them. At their most general level, information devices take one or more energy modulations as input and produce one or more forms of energy modulations as output. The model proposed by this paper further classifies information devices into four broad types:

- Processors,
- Routers,
- Memory components, and
- Communication components.

Processors transform a stream of input modulations into a stream of output modulations. Routers map a stream of input modulations to one or more streams of outputs. Memory components store input modulations and retrieve them when queried. Communication components move input modulations from one physical location to another. In a sense, communication components enable information to traverse physical space. Memory components enable information to traverse temporal space. Routers enable information to traverse topological space and processors enable information to traverse conceptual space. This paper will explore all of these notions in the following sections. At this point, the reader should note that this classification of information devices is strictly functional and independent of their specific implementation. As with information, the devices that support it can take many different forms (e.g., electronic components, neurons).

Assumptions 6 and 7 capture two more essential properties that all devices are assumed to have.

Assumption 6: Every information device occupies a unique and measurable position in physical space that no other device may occupy.

Assumption 7: Every information device occupies a finite, unique and measurable volume in physical space that no other device may occupy.

Assumption 6 implies that no two devices may occupy the same physical location and therefore a device's coordinates in physical space can uniquely specify it. Further, Assumption 7 implies that no point devices can exist (or, more accurately, can be modeled with this approach) and that no two devices may occupy overlapping volumes in physical space. This assumption captures a property associated with any practical device implemented with any existing technology.

Energy & Information Flows

The linkage between the flow of energy and information, described above, suggests a deep physical relationship that deserves further exploration. Figure 3 shows the energy flows through a general information device.

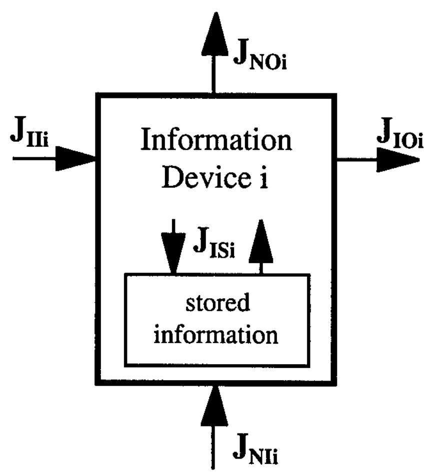


Figure 2. Model of the Energy Flows through a Generalized Information Device.

Since, from Assumption 2, information devices cannot defy the laws of thermodynamics, the conservation of energy suggests the following energy balance for an information device.

$$J_{IIi} + J_{NIi} = J_{IOi} + J_{NOi} + J_{ISi} \quad (1)$$

where J_{IIi} = energy flow into the device i through the flow of input information,

J_{NIi} = energy flow into the device i that does not contain information meaningful to the device,

J_{IOi} = energy flow out of the device i through the output information,

J_{NOi} = energy flow out of the device i that does not contain information meaningful to the device, and

J_{ISi} = energy flow into and out of the device i's local information store.

Equation (1) assumes that no information device stores energy that does not contain information relevant to it. This amounts to assuming that information devices do not store energy for later use (e.g., batteries) and that they have relatively low mass and heat capacity so they store no appreciable heat. Practical realizations of information devices can store consumable energy if their level of abstraction includes onboard batteries. While the proposed model excludes representing this sort of device explicitly, it can be modeled by using a separate representation for the energy storage component. In other words, this model requires that energy flows not contributing directly to the operation of the information devices must be modeled independently.

As a further comment on Equation (1), the storage term might seem to conflict with the partitioning of information device types since one could see that all devices can store information. In fact, this model assumes that only memory devices can store and retrieve persistent information, i.e., information that remains associated with the system even when

$$J_{NIi} = 0. \tag{2}$$

Thus, the proposed model permits all information devices to temporarily store information locally but only memory devices can store information when the energy flow to those devices is reduced to zero.

One can now specify the total energy flow required to sustain an information flow through the device by rearranging Equation (1) so that

$$J_{NIi} = (J_{IOi} + J_{ISi} - J_{IIi}) + J_{NOi}. \tag{3}$$

The term J_{NOi} really consists of two components so

$$J_{NOi} = J_{NDi} + J_{IEi} \tag{4}$$

where J_{NDi} = energy flow dissipated only because of the inefficiencies inherent to the device i and

J_{IEi} = energy flow dissipated only because of the erasure of information.

Substituting Equation (4) into Equation (3) gives

$$J_{NIi} = (J_{IOi} + J_{ISi} + J_{IEi} - J_{IIi}) + J_{NDi} \tag{5}$$

which separates the information bearing energy flow components from those without meaningful information.

Equation (5) permits comparing different information device technologies. If a device contains no inefficiencies that contribute to its energy dissipation then

$$J_{NDi} = 0 \tag{6}$$

and such a device approaches being adiabatic. Considerable research is ongoing to construct adiabatic logic devices [8-13]. Advances in this technology will reduce the amount of heat that computing circuits dissipate and that should simplify their packaging and reliability. However, for perfectly adiabatic devices

$$J_{NOi} = 0. \tag{7}$$

This requires that both

$$J_{NDi} = 0 \text{ and } J_{IEi} = 0. \tag{8}$$

If a device is perfectly energy efficient then

$$J_{NIi} = J_{IOi} + J_{ISi} + J_{IEi} - J_{IIi} \tag{9}$$

and if the device is perfectly adiabatic then

$$J_{NIi} = 0 \tag{10}$$

so that

$$J_{IOi} + J_{ISi} + J_{IEi} - J_{IIi} = 0. \tag{11}$$

Rearranging Equation (11) gives

$$J_{IEi} = J_{IIi} - J_{IOi} - J_{ISi}. \tag{12}$$

If the device stores no information locally then

$$J_{ISi} = 0 \tag{13}$$

and

$$J_{IEi} = J_{IIi} - J_{IOi}. \tag{14}$$

Therefore, a truly adiabatic information device must meet at least two conditions:

- Its function must be perfectly efficient; and
- Its logical operations must erase no information.

Landauer [14] has shown that a device can achieve the second condition if its logic is perfectly reversible. Fredkin and Toffoli [15], Landauer [14], Landauer and Bennett [16], Bennett [17, 18], and Feynman [19] have shown that logic can be constructed that meets this condition.

Equation (12) somewhat generalizes Landauer's result since it shows that logical reversibility (i.e., all information is conserved) is but one path to non-dissipative information devices. Other paths may also exist if those devices perfectly balance the energy flows output and into storage with those input and out of storage without preserving the information those flows represent. The logic of those devices need not be perfectly reversible, just their use of energy.

Landuaer's result leads to the conclusion that computing inherently requires no energy dissipation in its operation. However, as mentioned, this assumes that the logic executed in the computing operations is perfectly reversible. While Fredkin and Toffoli [15], Bennett [17], and Likharev [20-22] have shown how such devices could be constructed, Li and Vitanyi [29] have argued that using such logic to perform complex computing tasks may require significantly longer computing times to achieve the same result that irreversible logic produces.

Clearly, the logic circuits within today's computing systems are neither adiabatic nor reversible and so dissipate large amounts of energy. The amount of heat dissipated by present day computers illustrates this fact. However, the proportions of energy these practical devices dissipate resulting from information erasure and from device inefficiencies is currently poorly documented and, perhaps, poorly understood in general. This paper asserts that the energy flow dissipated due to device inefficiencies has two components such that

$$J_{NDi} = J_{N0i} + J_{NJi} \tag{15}$$

where J_{N0i} = energy flow the information device i dissipates independent of the information flow through the device, and

J_{NJi} = energy flow the information device i dissipates as a function of the information flow through the device.

Combining Equations (4) and (15) results in

$$J_{NOi} = J_{N0i} + (J_{IEi} + J_{NJi}) \tag{16}$$

where the term in parentheses represents the dissipation that depends upon the flow of information through the device.

Equation (16) suggests a way to partition the power into a device in order to assess its thermodynamic efficiency at one level. From Equation (16), the power dissipated by information device i is

$$P_{Oi} = P_{0i} + P_{Ii} \tag{17}$$

where P_{Oi} = total power dissipated by device i over time,

P_{0i} = power the device i dissipates when all information flows through it are zero, and

P_{Ii} = power the device i dissipates only as the result of information flowing through it.

In actual information devices implemented with today's technology,

$$P_{Ii} << P_{0i}. \tag{18}$$

This inequality can complicate the practical measurement of P_{Ii}. However, all devices for which Equation (18) is true will be terribly inefficient. As technology matures these inefficiencies will likely decrease.

From Equation (17) and thermodynamic definitions of efficiency, each device has an efficiency given by

$$\eta_{Di} = P_{Ii} / P_{Oi} \tag{19}$$

where η_{Di} = conversion efficiency for information device i.

The specific physical implementation of an information device determines its conversion efficiency. That conversion efficiency is largely independent of the content of the information flow through the device. If

$$\eta_{Di} << 1 \tag{20}$$

then the conversion efficiency can usually safely be assumed to be independent of the information flow rate.

Information Systems

Definition 3 proposes a meaning for the term information system as used in Definition 1.

Definition 3: Information system – an interconnected collection of information devices together with the energy modulations they contain.

This definition implies that all information systems have three properties in common:

- They consist of one or more of the information devices described above;
- Those devices are interconnected so they can interact with one another; and
- All information systems contain energy modulations that represent the information they possess.

The first property uses the notion of information devices to partition the functions of an information system into recognizable parts. This property could as easily say that all information systems manipulate, store, retrieve, distribute or communicate energy modulations. However, the notion of component devices enable these functions to be localized within the information system both topologically and physically. The second property describes information systems as a network of information devices. In this network, communications components link the other types together. Communication links connect both device types and their locations in physical space. A graph of processors, routers and memory devices connected through communication devices describes an information system's topology. This topological description abstracts implementation details away from the information system model.

The notion of devices also partitions an information system into its physical implementation (i.e., as a network of devices) and its information. In other words, the structures of information devices do not depend upon the content of the information they represent. This argument may not hold below certain levels of abstraction but it plays a key role in simplifying this model of information systems. This argument of independence enables viewing information systems as a network of devices (i.e., the information system's hardware) upon which information is superpositioned. Thus, the network of information devices create the medium within which information exists and information phenomena occur. The last property attempts to capture this character of information systems. This property suggests that the flow and storage of information completely describes any phenomena manifested by an information system. This property also strongly states that information systems require information, encoded as energy modulations, to exist. Without information, an information system's behavior can be explained completely by the behavior of the devices composing it.

Modulated Energy as Symbols

One can quantize energy modulations into symbols. This step takes information one level beyond its physical manifestation and closer to its existence as content. The character of information devices also changes at the symbolic level which permits describing the nature of their specific properties better. This results in further refinement of the basic device classes. Projecting information into symbols also provides the means for beginning to describe how the execution of information can produce information system behavior.

Symbolic Representations

The work of Turing [24], Shannon [5], Brillouin [25] and others organizes the energy modulations representing information into discrete symbols. For example, 0 and 1 represent two of the simplest symbols commonly used, especially for digital information systems. This permits discussing information independent of the particular encoding scheme that transforms energy modulations into symbols and, thus, provides one more level of abstraction through which to simplify complex information system behavior. As mentioned, information can exist through many different techniques of modulating energy (e.g., differences in amplitude, frequency and form). In fact, a single information system may represent information through many different forms of modulation. So, many mappings between energy modulations and symbols may exist within a single information system even though it may use only one common symbol set.

The reader must take care not to assume that any biases exist in this description toward electronic information systems, although these biases are quite common in other information system models. The linkage between energy modulations and symbols can take many forms. For example, the physical structures of ink on paper represent one form of energy modulation and the characters that those structures form represent meaningful symbols to at least one type of information system, humans. Many other diverse examples of symbology exist and the ability to represent information as collections of symbols appears to be an innate property of all information systems. This model consistently strives to employ levels of abstraction to preserve the essential properties of all information systems.

The diversity of symbology available and possible may pose, to the information system modeler, the problem of choosing the appropriate representation. After all, a complex information system may employ many different forms of symbology at different levels of content abstraction. The proposed model permits this to maintain its generality to all information system forms. As general guidance, one modeling a particular information system should choose a level of symbology that applies throughout the entire system being modeled. This choice will permit consistent and complete analysis without needing to represent multiple symbologies and the transforms that must exist between them. However, some cases may require representing information with different symbologies. For example, a system in which multiple languages represented the same concepts and where the translation between those languages is a process that contributes to the system's observable behavior. In this case, a critical part of the system's functions rest in the translation processes and these must be represented. In short, the proposed model gives the practitioner the flexibility to capture all of the important properties of an information system.

In general, the particular device implementation determines the form of encoding used although this encoding may also depend upon some information that the device uses. This commonly occurs in communications devices.

The partitioning of energy modulations into symbols also helps to make the proposed information system model consistent with the vast collection of models built upon information theory [5-7]. This conveniently avails all of the powerful tools developed within information theory to this information system model.

One can easily define a standard symbol that requires a well-defined amount of energy modulation to encode it and use that as a measure of information volume. This permits measuring information memory capability in standard symbols and information throughput in standard symbols/second. Henceforth in this discussion, use of the term symbol will assume the existence of such a standard symbol measure.

Information Devices & Symbols

As described above, all information devices modulate energy to encode information and each encoding represents an abstract symbol. The abstraction of energy modulations into symbols enables further differentiation and elaboration of information device functions.

This model represents processors as capable of taking one or more symbol streams as input and producing a symbol stream as output that generally differs from any of those received as input. Admittedly, cases could exist where a processor simply takes a symbol stream as input and produces an identical symbol stream as its output but the real power of processor devices comes from their inherent ability to perform symbol transforms. Only through executing these transforms can processors create new information within the system. In fact, this ability, which this model identifies with a single device type, captures the essence of all information system behavior when viewed simply as a black box (i.e., seen only through its mapping of input to output). Generally, when an information system consists of a single device, that device is a processor. This paper will give significantly more attention to processor behavior in later sections.

Routers take one or more symbol streams as input and reproduce one or more of those streams as outputs. Unlike processors, routers cannot change the symbol content or order of the input streams although they can select subsets of the input stream to route. While routers have significantly simpler behavior than processors, i.e., they simply appear to decide how to map input streams to output streams, they can have dramatic effect upon information system behavior, especially when that system consists of many components to which the symbol streams are routed.

Memory components support maintaining persistent information within an information system. Previous discussion of memory components only vaguely introduced their general function. More specifically, all memory components consist of two basic devices interacting through a memory medium:

- Memory writer and
- Memory reader.

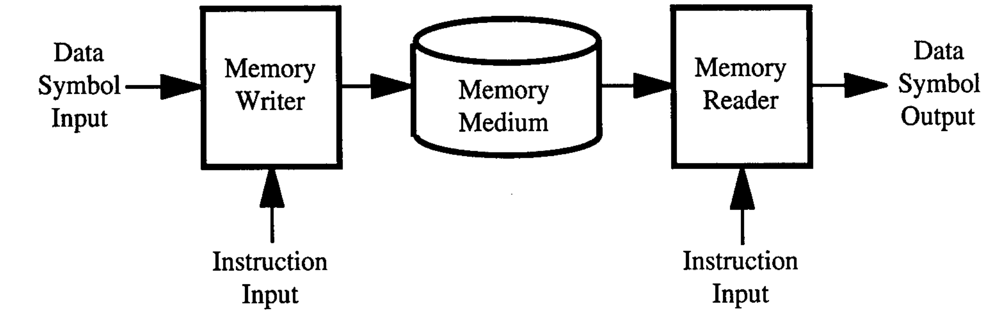
Figure 4 illustrates the configuration of memory components. Memory writers, hereafter called writers, transform input symbol streams into energy modulations of the memory media they access. Memory readers, hereafter called readers, detect the energy modulations storing information on the memory media and produce the symbols that represent those modulations. Readers can only produce symbol streams that writers have stored upon the memory medium they can access. However, readers do select the information stored on the medium that they produce as output. That selection is usually only a subset of all of the streams stored. In general, memory media contain fixed quantities of physical substances structured so that their potential fields can be spatially modulated and so that those modulations can be detected and distinguished so as to recover the information stored within the medium. Examples of memory media include magnetic and optical disks as well as parts of nonvolatile RAM. In this model, memory media are passive elements in that they cannot change their own state.

As discussed above, communication components move information from one physical location to another. They operate and are structured similarly to memory components. Like memory components, all communication components consist of two devices interacting through one medium, this one a communication channel:

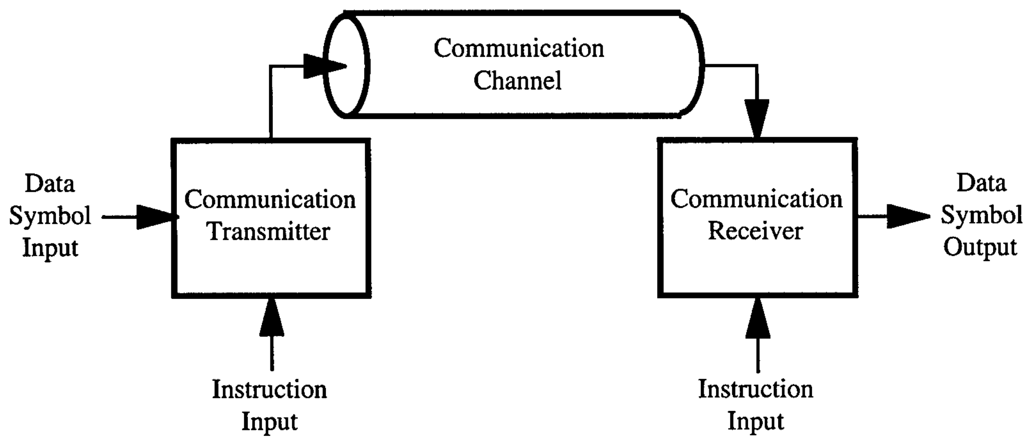
- Communication transmitter and
- Communication receiver.

Figure 4 illustrates the configuration of communications components. A communication transmitter, hereafter called a transmitter, transforms input symbol streams into energy modulations of the communication channel it accesses. A communication receiver, hereafter called receiver, detects the energy modulations of the communication channel it accesses and produces the symbols that represent those modulations. Receivers can only produce symbol streams that transmitters have transmitted on the communications channels they share. A communication channel transports information from all of the transmitters coupled to that channel to all of the receivers coupled to that same channel.

Configuring a communication component as a transmitter sending information to a receiver through a communication channel incorporates the model proposed by Shannon and Weaver [5] for a communication system. This takes another step to access the modeling mechanisms developed in information theory [5-7].



Memory Component Configuration



Communication Component Configuration

Figure 3. Memory and Communication Component Configurations.

In summary, at the symbolic level, one can compose an information system as a network of six device types:

- Processors,
- Routers,
- Memory writers,
- Memory readers,
- Communication transmitters, and
- Communication receivers

interacting through two media types:

- Memory media and
- Communications channels.

Information, as quantized as symbols, flows through these devices and the associated communications channels and is stored in the memory media. One can combine the primitive information devices just described into components that correspond more directly with the components seen in complex large scale information systems. Figure 5 illustrates five basic types of information devices and three ways in which these devices can be combined into meaningful components.

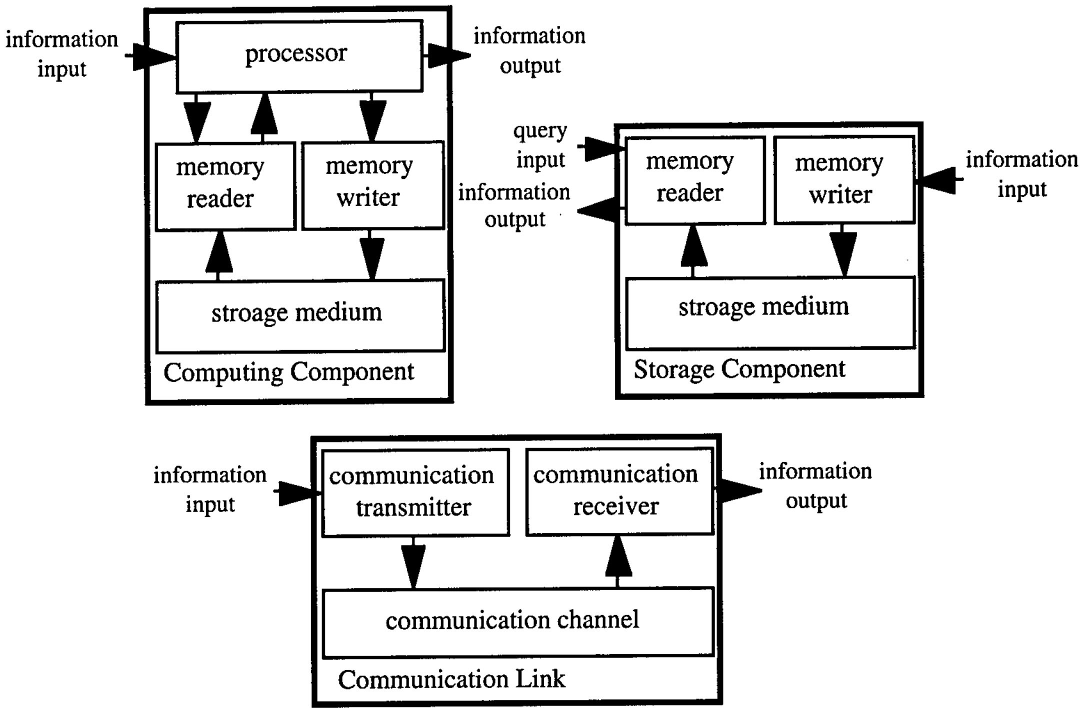


Figure 4. Common Compositions of Primitive Information Devices.

Quantizing the energy modulations representing information into symbols effectively creates separate identifiable flows through an information device from their energy flows. Figure 6 depicts a generalized model of the information flows within an information device.

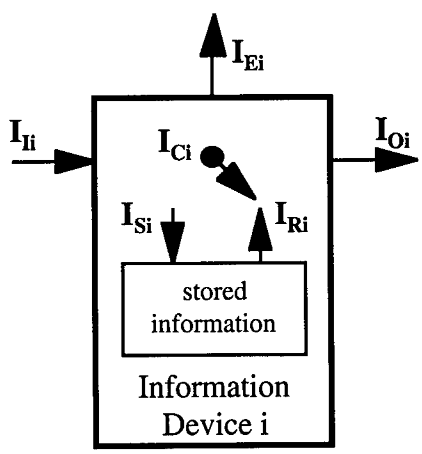


Figure 5. Information Flows through a Generalized Information Device.

Like energy flows, one can construct a conservation equation describing the information flows within an information device.

$$I_{Ii} + I_{Ci} + I_{Ri} = I_{Oi} + I_{Si} + I_{Ei} \tag{21}$$

where I_{Ii} = information flow into the device i,

I_{Ci} = information flow created by device i,

I_{Ri} = information flow retrieved from local storage within device i,

I_{Oi} = information flow out of the device i,

I_{Si} = information flow stored to the local information storage within device i, and

I_{Ei} = information flow from information erasure within device i.

Unlike Equation (1) describing the conservation of energy for an information device, this conservation equation must include flows to account for information creation and erasure. In other words, no thermodynamic law constrains an information device from creating or destroying information as it does for energy. Further, the units calibrating information flows differ from those representing energy flows. The units of energy flows are amount of energy (e.g., erg, Joule)/sec where the most basic units of information flows are commonly symbols/second.

Symbol Execution

Within every information system, there exists a subset of its symbols that can be executed. Executable symbols represent an information system's behavioral information. In this model, executable symbols are called instructions. Definition 4 defines the term instruction in the context of an information system model.

Definition 4: Instruction – a symbol that at least one device in an information system interprets to change that operations that it performs upon its input symbol stream (i.e., an executable symbol).

Thus, this model partitions symbols into two disjoint sets, executable and nonexecutable symbols. An instruction tells an information device how to treat the symbols of its input stream. By associating executable symbols with the notion of instructions, a device can execute any symbol that is a member of its instruction set. The instructions input to a device define the transforms that device performs upon its input symbol streams and, therefore, determines the change in information content they make. Unfortunately, the association of instructions with devices has historically been reserved for modeling the behavior of processors. The model in this paper permits any information device to have an instruction set. In the most general sense, all information devices can execute part or all of the symbols that compose their input information streams. Thus, each device within an information system can effectively accept two distinct streams of input symbols, one representing its instruction stream and the other its input data stream.

Clearly, each device type interprets symbol execution somewhat differently. Table 1 summarizes these interpretations for each information device that this paper has defined.

Table 1. Interpretations of Executable Symbols for the Different Information Device Types.

Information Device Type	Executable Symbol Interpretation
Processor	Element of the transforms performed upon the input data symbol stream to produce the output symbol stream
Router	Element of the transforms that determine from the content of the input data symbol stream the paths to which the input data symbol stream should be routed
Writer	Element of the processes for organizing the information upon accessible memory medium
Reader	Element of the query for accessing the information on the accessible memory medium
Transmitter	Element of the process for encoding the input symbol stream into modulations of the accessible communication channel
Receiver	Element of the process for decoding the modulations of the accessible communication channel into the output symbol stream

The interpretation of instructions for processors is consistent with traditional interpretations. Processors execute their instruction streams with input data streams and produce data output streams through these executions. Memory writers essentially map information into the state space that the memory medium provides. Since many different mappings are possible, instructions tell a writer which mapping to choose, a choice that may depend upon the content of the input data symbol stream. Similarly, an instruction tells a memory reader what information to extract from the information structured in the accessible memory medium. In this case, an instruction can be as simple as an address (i.e., such as one would see in a processor accessing contents of its nonvolatile RAM) or as complex as an elementary relationship of a database query. Instructions for transmitters and receivers dictate encoding and decoding actions. Thus, two conditions must be met for a transmitter and receiver to interact without information loss:

- A common communication channel must exist between them; and
- The receiver's decoding scheme must be the exact inverse of the transmitter's encoding scheme.

Here, instructions represent the encoding and decoding schemes. This capability enables efficiently representing such processes as encryption and error detection and correction coding as well as various modulation variances that transmitters may use for different purposes (e.g., spread spectrum, frequency hopping). The constraints for communication also accommodate the description of various lossy encoding schemes.

All of the symbol streams input to memory readers and communication receivers are assumed to be instructions since their input data streams come from the accessible memory media and communications channels. The input data streams of processors, routers, memory writers and communications transmitters must be divided into instructions and nonexecutable data streams.

The execution of every instruction changes the state of the information system. Therefore, instructions take the first step toward representing the effects that information can have upon information system behavior. Although using instructions to capture information system behavior corresponds most closely with the well known von Neumann model of processing [26], the model of executable symbols corresponds with some interpretations of Turing machines [24, 27]. Enabling every device type to respond to instructions generalizes the device models and makes those models more efficient than those available through more primitive representations that restrict execution to a single device type. Of course, this flexibility comes at the price of slightly more complex model primitives.

In the proposed information device model, symbol execution drives the information flows through the device. Assumption 8 captures this aspect of the proposed model.

<u>Assumption 8:</u> Any flow of symbols across a device's boundaries (i.e., information into or out of the device) or within a device (i.e., information stored, retrieved, created or erased) requires the execution of at least one symbol.

For a device that performs the same operation repeatedly, the instructions representing that operation may be embedded in the device's hardware implementation. On the other hand, programmable devices (e.g., microprocessors) may execute the symbols within a separate information flow stream.

The proposed model further connects energy flow with symbol execution as described in Assumption 9.

<u>Assumption 9:</u> Every information device performs some amount of physical work each time it executes a symbol.

From the previous discussion on energy flows in information devices, this assumption limits the applicability of the proposed device model to those for which

$$J_{NI} \neq 0. \tag{22}$$

This means that this model applies only to those devices that meet one or more of the following conditions:

- Dissipate energy as a function of each operation performed,
- Erase information with each operation performed, or
- Irrecoverably store information with each operation.

This model currently encompasses all practical information devices. As mentioned, research is currently ongoing to address one or more of these conditions. When that research results in practical information devices that meet none of these conditions then the proposed information device model will no longer generally apply unaltered.

Assumption 9 enables assigning to each instruction a measure of the work that a specific device must invest to execute that symbol. This measure depends both upon the instruction and the implementation of the device executing it. Normalizing the work required to execute a symbol can help to separate the effects of implementation so that

$$W_{ij} = \eta_{Sij}\, w_j \tag{23}$$

where W_{ij} = the work device i invests to execute symbol j,

η_{Sij} = the efficiency with which device j executes symbol j, and

w_j = the normalized work required for any device in an information system to execute symbol j.

This normalization process assumes that one knows in advance the amount of work required for every device to execute the executable symbols to which a system may be exposed.

Device & Media Properties

The characterization of information devices presented above suggests several properties by which to describe both the state and capabilities of those devices. Table 2 lists those device properties whose state can vary over time. The instantaneous values of these properties describe a device's state. Table 2 also suggests the units through which to measure those properties.

Table 2. Information Device Properties Assumed Dependent upon Time.

Property	Definition	Measurement Units
Physical Location	a unique and measurable position in physical space relative to a chosen origin and a chosen point within the device's volume	(length, length, length)
Input Information Flow Rate	the rate at which meaningful information enters the device	symbols/unit time
Output Information Flow Rate	the rate at which a device produces information as output	symbols/unit time
Stored Information Flow Rate	the rate at which a device stores information in its local memory	symbols/unit time

Table 2. Information Device Properties Assumed Dependent upon Time (continued)

Property	Definition	Measurement Units
Retrieved Information Flow Rate	the rate at which a device can retrieve information stored in its local memory	symbols/unit time
Consumed Information Storage	the total amount of the local memory consumed by stored information	symbols
Information Execution Rate	the rate at which a device executes its instructions	standard instructions/unit time
Total Information Volume	the total amount of information that a device represents at one instant in time	symbols
Execution Work Rate	the rate at which a device performs execution work at one instant in time	energy/unit time

Table 3 describes those properties of an information device which may be assumed to remain constant over time. These properties describe the capabilities of the device. Table 3 also suggests the units by which to characterize these properties.

Table 3. Information Device Properties Assumed Independent of Time.

Property	Definition	Measurement Units
Physical Volume	a unique, finite and measurable volume in physical space	$length^3$ (e.g., cm^3)
Input Bandwidth	the maximum rate at which meaningful information can enter a device	symbols/unit time
Output Bandwidth	the maximum rate at which a device can produce information as output	symbols/unit time
Storage Bandwidth	the maximum rate at which a device can store information in its local memory	symbols/unit time
Local Memory Capacity	the maximum local memory capacity that a device maintains	Symbols
Total Information Capacity	the total amount of information that a device can represent at any instant in time	symbols
Execution Work Capacity	the total rate at which a device can perform execution work	energy/unit time

The two memory devices, writers and readers, require an additional property each to account for their interactions with their associated memory media. A memory writer has a property called write latency, the time interval from when an encoded symbol begins to be written onto a memory medium until that symbol is available to be read by any reader that can access that same medium. Correspondingly, a memory reader has a property called seek latency, the time interval from when the reader identifies a the location in the medium from which it intends to read until the symbol at that location is accessible to the reader. Write and seek latencies may only apply to certain common memory mechanisms such as rotating magnetic disks and they may depend upon the state of the mechanism and the location of the available capacity, for write latency, or existing symbol, for seek latency. This model assumes that the values of both properties depend upon time and that their measurement units are in time.

The device properties listed in Tables 2 and 3 are by no means comprehensive. They do represent the level of abstraction assumed by the proposed model. All of the information flow rates and bandwidths derive from the device models presented in Figure 7 and Equation (21). Table 3 implies an important assumption of the proposed model and captured below.

Assumption 10: Every information device has a limited capability to perform its functions.

Besides limited bandwidths for every information device, Assumption 10 limits a device's capacity to perform execution work. This limitation provides one mechanism through which to model a device's latency. Later sections in this paper will explore the connection between latency and execution work capacity in more detail.

Like information devices, the media that connect writers to readers and transmitters to receivers also have properties and capability limitations. Tables 4 and 5 summarize these.

Table 4. Information Medium Properties Assumed Dependent upon Time.

Property	Medium Type	Definition	Measurement Units
Occupied Information Storage	Memory	the total amount of the medium currently occupied by information	symbols
Information Flow Rate	Communication	the total amount of a channel's bandwidth occupied by information traffic	symbols/unit time
Symbol Propagation Time	Communication	the time interval from when a symbol is transmitted into the channel until that symbol is received by a receiver	time

Table 5. Information Medium Properties Assumed Independent of Time.

Property	Medium Type	Definition	Measurement Units
Physical Volume	Memory	a unique, finite and measurable volume in physical space	$length^3$ (e.g., cm^3)
Total Information Capacity	Memory	the total amount of information that a memory medium can contain	symbols
Channel Bandwidth	Communication	the total information flow rate that a channel can sustain between transmitter-receiver pairs	symbols/unit time

In this model, a memory medium occupies a unique volume within physical space where a communication channel does not. Further, a symbol's propagation time may vary with time to account for changes in the relative positions of transmitter-receiver pairs. However, this model assumes the channel capacity to be constant. This could easily be modified to account for dynamic channel bandwidths in future enhancements.

Tables 4 and 5 imply some important assumptions about the behavior of memory and communications media.

Assumption 11: Every symbol occupies a finite fraction of the capacity of a memory medium or communication channel.

Assumptions 10 and 11 suggest that for a symbol representing some information to exist within a device or medium, it must consume some fraction of the capacity of that device or medium. In a sense, each symbol occupies a fraction of the device's or medium's capacity space. This leads to another assumption contained by the proposed model.

Assumption 12: No two symbols may occupy the same part of the capacity of a device or medium.

Assumption 12 constitutes the equivalent of the Pauli exclusion principle for information devices and media. It, post facto, articulates the reason for why each device and medium has a limited capacity to store symbols and support symbol flows. Admittedly, communications channels present somewhat of a special challenge since the symbols within them do not necessarily occupy a unique physical volume. They do, however, occupy a unique and finite part of the state space associated with the channel. As a result of Assumption 12, imposing one symbol onto the space occupied by another symbol irreversibly destroys the existence of the precedent symbol and loses the information represented by that symbol. This implies that if a device's or medium's capacity is completely consumed then forcing additional input into that device or medium will necessarily and irreversibly destroy some of the information it represents.

Since every device and memory medium occupies a unique and finite volume in physical space and they have limited capacities then each symbol must also occupy a specific physical volume at a specific physical location. This notion of information volume is independent of the content of the information encoded.

Symbols as Concepts

Recognizing the universality of object-oriented descriptions provides the mechanism to abstract symbol collections into basic concepts. Further quantizing information from symbols into meaningful concepts enables connecting basic device behavior to the effects that behavior has upon information content. It also provides the means to describe the fundamentals of information complexity and the reason for one form of information flow within a system, information diffusion.

Conceptual Information

Definition 2 articulates an essential property of information content, meaning to an information system. Symbols individually do not usually represent information content. The proposed information system model assembles content from collections of symbols. These concepts represent the essence of the information content an information system contains and uses to determine its behavior. The proposed information model partitions all information content as described by Assumption 13.

Assumption 13: All information systems represent all of their information content as abstract objects and describe those objects in terms of their state and behavior.

Definition 5: Abstract object - an object that an information system represents that could (1) exist elsewhere, perhaps as real physical objects or as other abstract objects represented within another information system, and (2) represents only a subset of the state and behavior of the actual object.

From Definition 5, an abstract object only approximates the state and behavior of the object it represents. The process of deciding what to represent and what not to represent is called abstraction. The abstraction process produces abstract objects. All information content embodies abstractions, sometimes very precise approximations of the real objects but always approximations. The objects about which these abstractions are constructed may exist as physical objects or they may exist solely as abstract representations within some information system. Pretty well understood physical laws constrain the objects in the physical world but no such constraints seem to exist for the objects of the abstract world.

Information systems use collections of symbols to represent object state and behavior. A symbol collection representing object state describes of a set of one or more properties and a collection representing object behavior describes a set of one or more executable dependencies. State representations consist primarily of data symbols.

The properties of the objects that an information system represents completely describe the state of those objects within the context of what that information system represents. An object-property-value tuple effectively represents an object property concept. This tuple

- Identifies the specific object with which the property is associated,
- Identifies the specific property being described, and
- Describes the state of that property at the time.

Assumption 14: Each property associated with a specific object is unique within the context of that object.

Assumption 14 permits an object's set of properties to uniquely distinguish that object from all the others represented within an information system. This leads to the following definition of object equivalence.

Definition 6: If two objects exist within different parts of an information system such that

- Those objects have identical sets of properties and
- The values of those properties have equal values

then those two object representations are equivalent.

Multiple equivalent representations means that the information system maintains multiple instantiations of the same objects. The physical world generally prevents the existence of non-unique objects. This constraint does not apply within the world maintained in an information system. In fact, practical information systems often proliferate multiple representations of the same content, sometimes for very good reasons such as redundancy and other times out of laziness.

Assumption 14 further implies that no object represented by an information system can have two properties that represent the same state. This seems somewhat an artificial constraint but it helps to distinguish multiple instantiations of the same object.

Executable dependencies, assembled from instructions, completely describe the behavior of these objects by changing their property states. Without executable dependencies between object properties, object state would remain static and the information system representing those objects would manifest no observable behavior. Definition 7 describes the nature of an executable dependency.

Definition 7: An executable dependency, or simply dependency,

- Is a finite set of instructions that can all be executed upon a single device,
- Produces a finite set of output symbols from a finite set of input symbols, and
- May include a set of nonexecutable symbols that describe constant state upon which the dependency depends.

Each object dependency couples one object state to another. The execution of each dependency can change the information content represented within an information system and thus creates the observable behavior that system manifests. However, object dependencies cannot exist simply as abstractions. When a processor executes a dependency, the input data stream to that processor represents the independent state variables and the output data stream that processor produces represents the dependent state variables. Dependencies can contain both executable and nonexecutable data symbols but the instructions represent the primary constituent of all dependencies. The nonexecutable symbols define the parameters upon which the instructions depend when executed. These parameters can represent the values of constant object properties. In some cases, dependencies can take other dependencies as input or produce dependencies as output (e.g., dependencies representing compilers). In these cases, the dependency descriptions represent both object state and behavior but in different contexts. In essence, a dependency defines a transform from one abstract state to another and when that transform is executed the state of the information content within an information system changes. Each dependency can be completely defined by an execution graph consisting of a set of instructions and the set of all executable transitions between them. Definition 8 describes the conditions necessary for a device to execute a dependency.

Definition 8: Executable dependency – a dependency whose execution graph is completely contained by the execution graph of at least one device in the information system.

This description of executable information links to information physics to the body of computational theory [27, 28]. Dependencies begin to define the nature of and necessity for executable information. Thus, object dependencies represent the executable information contained at the information system level. Since dependencies must be executable by the system's devices, instructions must compose object dependencies. At least one instruction must be executed for every dependency to transform an input concept into an output concept.

Concepts & Information Devices

Except for memory devices, the primary interactions between concepts and devices arise from the execution of dependencies. Since every information device type can potentially execute an input symbol stream, potentially every device type can execute dependencies input to it. However, the different device types interpret dependencies differently. Table 6 describes the different interpretations different device types have for dependencies.

For a processor, an executable dependency is like a program or components of a program such as a function. Many dependencies representing single functions can be organized into a single concept equivalent to a computer program. A processor executing a dependency may change the state of an existing abstract property instance, create a new instance of an abstract property or delete the instance of an abstract property. Memory readers treat the dependencies they execute as queries. Communication transmitters and receivers use dependencies to access particular channels to which they may have access and deal with different information encodings. Similarly, memory writers execute dependencies to direct the incoming information to particular locations in the medium's memory space.

Table 6. Interpretations of Executable Dependencies for the Different Information Device Types.

Information Device Type	Executable Dependency Interpretation
Processor	A transform performed upon the input data stream that produces the output data stream
Router	A transform that determines from the content of the input data stream the paths to which the input data stream should be routed
Writer	A transform that maps the input data stream into the accessible and available memory medium capacity
Reader	A query for accessing the information on the accessible memory medium
Transmitter	A transform for encoding the input data stream into modulations of the accessible communication channel
Receiver	A transform for decoding the modulations of the accessible communication channel into the output data stream

Since all dependencies are composed of instructions then dependency execution also requires physical work to be performed. Each dependency can be defined by a factor that contributes to computing the amount of work required to achieve a specific result through the execution of the dependency where

$$W_{Di} = N_{Di}\, W_{Ii} \tag{24}$$

where W_{Di} = work required to execute dependency i, and

N_{Ii} = number of standard instructions executed per dependency.

Further,

$$N_{Di} = V_{Ii}\, N_{DPi} \tag{25}$$

where V_{Ii} = volume of input data, and

N_{DPi} = number of standard instructions executed per unit volume of input data.

The total amount of work contained by a dependency depends upon the data input volume because a single dependency might be executed many times on a single input symbol to achieve the desired output information.

The notion of information content conservation becomes important at the concept level. Definition 9 describes the meaning of content conservation in the context of the proposed model.

Definition 9: Concept conservation – the condition when a device can completely preserve the meaning of a concept to an information system through the transform it executes.

At a minimum, an information device conserves information if and only if the symbols it produces as output are a proper subset of the set of symbols to which it has access from the medium that provides its data input. For example, when a transmitter employs a lossless encoding scheme and the receiver of its transmission employs a decoding scheme that is the exact inverse of the transmitter's encoding scheme then the communicated concept is guaranteed to be conserved (i.e., assuming that a lossless communication channel) through the communication component. Therefore, communication devices tend to conserve the information content with which they deal. Similarly, the concepts a memory writer writes to the memory medium should be completely recoverable by any memory readers that have access to that same medium. Thus, memory components also tend to conserve the information with which they deal. Content conservation also implies that a communication receiver and a memory reader only have access to the content produced by the transmitters and writers with which they share media. They cannot, of their own, produce new information content.

By Definition 9, routers also tend to conserve information. Of all the information device types described above, only processors do not conserve information content. In fact, processors specifically try not to conserve information. As a result, new information to an information system, other than that received as input, can only come from processors. Further, their reliance upon executing dependencies for their function means that processors can only produce new information if they execute dependencies that result in new states or state changes within the system. This notion localizes new information production to only the processors within an information system.

Information System Complexity

Structuring an information system's information content as concepts representing object properties and dependencies facilitates describing information system complexity. Information theory essentially deals with information independent of its content [6, 25] but the physics governing abstract worlds depends strongly upon that very content. Feynman [19] suggests a definition of information content based upon thermodynamic interpretations but that definition only links information to the state of a physical system. Bennett [29, 30], Zurek [31, 32], and Li and Vitanyi [33] have all proposed various interpretations of information content and try to explain the effects of that content upon macroscopic information system behavior. All of these interpretations of information content and of the relationship between information content and information system complexity have their weaknesses. Further, none of these descriptions of information content is widely accepted or practically applied to information system design.

The proposed model approaches the problem of information content and complexity from a slightly different perspective, one that may lead to better acceptance and practical application because of its broader and more macroscopic interpretation. Clearly, the existence of information within a system can contribute to that system's behavioral complexity. In this model, information content represents the meaning that information has to the information systems with which it interacts. Thus, the meaning of information content to a system is directly proportional to the number of different ways that the system uses that information to shape its behavior. As the amount of meaningful information within a system increases and the amount of meaning that has to the system increases, the complexity of the system's behavior increases monotonically.

These observations suggest that the complexity of an information system is proportional to the number of

- Dependencies that system possesses, and
- Ways those dependencies interact with the state information maintained by the system.

This suggestion casts the notion of complexity into the terms of object properties and dependencies. Thus, the proposed model asserts that one simple measure of information system complexity contributed by the information content to which the system responds can be approximated by the relationship

$$C_{SI} = \Sigma\, \Sigma\, N_{Pij}\, N_{Dij} \tag{26}$$

where C_{SI} = complexity of an information system created by the existence of meaningful information within that system,

N_{Pij} = number of object properties that specify the independent variables of the jth dependency of object instance i, and

N_{Dij} = number of object properties that specify the dependent variables of the jth dependency of object instance i.

Equation (26) requires counting the dependencies between each instance of every object class. This implies that if a system represents no instances of an object for which it propagates behavior then any information pertaining to that object does not contribute to that system's complexity.

Complexity is necessarily a macroscopic property that really arises from deeper statistical arguments. Complexity, as described by Equation (26), measures the total number of dependencies that could be executed to take the system from one state to another where the state of the object concepts a system's information content represents define the information system's instantaneous state. Further, the complexity of an information system depends upon the fidelity of the information content it represents [34]. Information complexity really measures the size of the possible behavior space of an information system. In effect, complexity measures amount of information content that an information system possesses. This formulation appeals to the intuitive notions that information system complexity increases monotonically with the number of dependencies the system possesses and the number of different data items upon which those dependencies can operate. This is equivalent to the number of objects the system represents and the number of ways those objects can interact.

Equation (26) only describes one component of an information system's complexity, that contribution created by the system's information content. In fact, the total complexity of an information system consists of two components such that

$$C_S = C_{SI} + C_{SD} \tag{27}$$

where C_S = total complexity of an information system and

C_{SD} = complexity contributed by the devices and media the system contains.

Any information an information system produces in the course of its functions contributes to that system's complexity if

- The resultant information is either stored or communicated directly to a device executing a dependency that responds to that information,
- Dependencies exist that could respond to that information, or
- That information represents a new dependency that depends upon information that exists within the system.

Many [25, 32, 33, 35-43] have argued for a role for physical entropy in contributing to information system behavior. The model proposed in this paper concurs but goes farther to suggest that complexity and physical entropy are related.

Informally, complex systems, in the limit, can be made to appear completely disordered by simply increasing the number of dependencies coupling their observable states. Further, the degree of disorder that a complex system expresses appears to increase as the complexity of that system increases. Thus, in the limit no difference appears to exist between a completely disordered system and an extremely complex system. These informal observations suggest that the physical entropy of a system, a measure of its disorder, seems to be a monotonically increasing function of its complexity. This observation suggests that the complexity information content contributes to an information system should also contribute to that system's physical entropy. The definition of information complexity provides a path to specifically describe that relationship.

In effect, CSI describes the total number of paths that exist to take an information system from any state to any other state in its behavioral space. By approximating that for any state i, the number of paths that take the system to that state are equally likely (a big big assumption) then the probability that the system will be in state i is given by

$$p_{Ii} = C_{SIi} / C_{SI} \tag{28}$$

where p_{Ii} = probability that a system's information state will be in state i and

C_{SIi} = total number of dependencies whose execution could take the system to state i from any other state.

From the well known definition of entropy from statistical thermodynamics [44], the entropy of a system is

$$S = -k \Sigma p_i \ln p_i \tag{29}$$

where S = system entropy,

p_i = probability of reaching system state i, and

k = Boltzmann's constant.

Combining Equations (28) and (29) leads to the following relationship between the physical entropy contributed by an information system's content and its complexity.

$$S_{SI} = -k \Sigma (C_{SIi} / C_S) \ln (C_{SIi} / C_S) \tag{30}$$

where the summation is performed the entire state space described by the object properties that the system's content represents.

Intuitively, Equation (30) makes sense since the entropy of a system depends both upon the number of dependencies that it can execute and upon the size of the state space that its content represents. It also says that information that does not explicitly contribute to the information system's behavior (e.g., state information that no dependencies involve in their independent and dependent variables, and dependencies that are not executed because no states exist involving them) contributes nothing to the system's entropy and apparent disorder. But, Equation (30) relies upon the key assumption that all dependencies are equally likely to be executed and this assumption is clearly not true for most information systems. However, Equation (30) does provide an upper limit upon the contribution a system's information makes to its physical entropy.

If complexity and entropy are related, as shown in Equation (30) then complexity (since it is a component of entropy) must obey the Second Law of Thermodynamics or

$$\partial (C_S)/ \partial t \geq 0 \tag{31}$$

for any closed information system.

The Second Law as applied to a system's information content appears to state that the complexity of that content will always increase or remain the same. This pressure to increase system complexity represents another source of the forces that drive information flows within an information system. This leads to the conclusions that, if uncontrolled, the complexity of an information system will increase until its information consumes all of its capabilities.

Further, a state of absolute zero completely lacks executable information. Thus, for any information system,

$$C_S > 0 \tag{32}$$

for that system to possess any information responsible for its behavior. Equation (32) also suggests that information contributes to system complexity if and only if it consists of dependencies or it is pure data upon which its dependencies depend.

Information processing can change the total complexity of an information system. Information complexity also appears to be related to the amount of computation required to achieve a certain result. This would relate the complexity changes in an abstract world to the information flows through the information devices. Information complexity changes require execution work to be performed. Equation (26) suggests that it takes energy to change the entropy of a system. Thus,

$$W_{Di} = f(\Delta C_I) \tag{33}$$

where ΔC_{Di} = complexity change made to the system's information in performing that execution.

Equation (33) can be linearly approximated by

$$W_O = a_J V_D + a_W \Delta C_{Di} \tag{34}$$

where W_O = work required to produce an output,

a_J = constant that relates data flow to work,

V_D = data flow volume required to produce the output, and

a_W = constant that relates the units of complexity change to the units of execution work.

The first term of Equation (34) accounts for the work required just to sustain a data flow through an information device for the time needed to produce the desired output. The second term in Equation (34) accounts for the work required to make the complexity change.

Concepts as Information Content

The final abstraction step in this model partitions object state into actual and desired state. This introduces the mechanism through which to describe the intent of information systems. This mechanism describes the phenomenon underlying the flow of most information within a system. Finally, all of the previous concepts are tied into a description of integrated information system behavior.

Abstract State Partitions

The object state an information system's content represents can be further partitioned to improve the resolution of the proposed model's descriptions. Assumption 15 describes this partitioning.

Assumption 15: Any object state within an information system can simultaneously assume two distinct values where one represents the actual or probable state of the property and the other represents the property state desired by an information system.

The actual state represents object state drawn from direct observations or inferred from those observations. This represents information about the state of the corresponding real object in the past or the future. Since all acts of observation and inference require a finite amount of time to perform, all representations of present object are considered to be past states albeit recent past. All knowledge of future probable state result from inferences of observed or inferred past state and represent the system's predictions of real object behavior. This implies that execution work must be performed to obtain this knowledge.

Information about desired object state represents a system's intent and places the notion of object state into a direct relationship with the reasons for the information system's existence. This concept clearly relies upon some assumptions that have complicated philosophical interpretations. However, this paper simply asserts that all information systems exist for some purpose and through that assertion circumnavigates those political shoals. A system's information content describing the desired state of the objects it represents are its goals. Goal states, described in terms of object-property-value tuples, represent those states that the system desires to achieve. The actions of an information system with goals attempt to change the real world state to achieve those goals. In effect, goals are a special type of information since systems that possess goals respond to them by trying to achieve them. Thus, the goals of an information system describe its purpose. Goals exist as data but goals can drive device behavior as data input. However, goals must provide input to executing dependencies for them to propel any information fluxes. Goal-driven dependencies have at least one goal property as an independent variable.

This final brick in the information content castle permits structuring an information system into three distinct layers that shape the nature of the system's behavior:

- Device layer – that layer that establishes the capability limits of the information system and so defines the boundaries of the system's behavior.
- Information layer – that layer that creates the load upon the device layer and provides the medium within which the content exists.
- Content layer – that layer that defines the specifics of the objects represented as well as specifying the reasons for why the devices function at all.

Information Flow Mechanisms

Information flows within all information systems. The very behavior of information systems depends upon the information that flows within them. This observation suggests information flows as an intrinsic characteristic of information systems, a fundamental observable phenomena. This section proposes two primary mechanisms underlying information flows and quantifies their dependencies.

In all thermodynamically-constrained systems, the performance of physical work requires the existence of some form of forces. Since all information flow requires work to be performed then some forces must exist that drive those flows. Therefore, some conditions must exist within information systems that manifest the forces that drive the changes in that system's abstract world state. In this context, information flows include all information movement into and out of all information devices, into and out of the memory media, through the communications channels, information erasure, information creation, and instruction execution. Since the structures at the device and information layers of an information system do not, by themselves, cause information to flow within that system, one can further suppose that the origin of those forces resides in the content layer. The previous discussion on information diffusion describes one form of such a force but observations of existing information system behavior suggest that more powerful forces must exist since diffusion alone can explain only a relatively small proportion of all observable information flow.

Empirically, the existence of goals seems to cause information flows and so could be another source of forces that drive these flows. As described, goals are a special kind of declarative information that information devices can possess and use. An information system's behavior cannot reflect the influence of a goal unless it explicitly represents the goal state information describing that goal. Goals can either pull information from information sources or push information to information sinks. These observations lead to Assumption 16.

<u>Assumption 16:</u> Neglecting the information flows from entropic forces (see the discussion below), no information flows through a device without the existence of one or more goals.

Assumption 16 tacitly states that information flow rate depends upon the forces that the existence of goals create or

$$I_{IGi} = f(E_G) \tag{35}$$

where E_G = magnitude of the force exerted by information system goals and

I_{IGi} = total information flow through device i caused by the existence of the information system's goals.

One approach to identifying the nature of $f(E_G)$ employed in the physical sciences approximates this function through a power series and neglecting the higher order terms. Since No information flows when no goal exists the constant term of the series is zero and

$$I_{Ii} = a_{1i} E_G \tag{36}$$

where a_{1i} = a constant relating goal force to information flow rate.

Keeping with the separation of the effects of information devices and information content, the constant a_{1i} depends upon device characteristics. Incorporating this notion into Equation (36) and rearranging it gives

$$E_G = I_{Ii} R_{Ii} \tag{37}$$

where R_{Ii} = device resistance to information flow and thus $R_{Ii} = a_{Ii}^{-1}$.

Equation (37) relates a physical observable (i.e., information flow) with the abstract state of representing goals and the means to achieve them, an element of information content. Goals must exist (i.e., be represented within the system) and be used as input to the object dependencies for an information system to manifest their effect. Goals appear to drive most information flows within information systems, particularly in organizations. Embedded in Equation (37) is the assumption that the relationship between goal force and information flow rate is simply linear. While a more complex relationship may exist, linear seems like a good place to start especially since this relationship must be applied to all of the components of an information system thereby creating a very large system of interdependent linear equations that must be solved.

The notion that information devices perform physical work, proposed earlier, provides the means to compute the factor R_{Ii} for an information device by measuring the information flow and the work the device performs. Since the power a device dissipates is given by

$$P_{Ii} = E_{Ii}\, I_{Ii} \tag{38}$$

And, recalling that the relationship between work and power [45] is

$$W_{Ii} = P_{Ii}\, \Delta t \tag{39}$$

where W_{Ii} = work performed by the device i and

Δt = the time interval over which the information flow is sustained.

then substituting Equations (37) and (37) into Equation (39) obtains

$$W_{Di} = P_{Di}\, \Delta t \ = I_{Ii}^{2}\, R_{Ii}\, \Delta t \tag{40}$$

If all of the energy that a device dissipates results from the information flows through it and all of those flows are driven by goals then Equation (40) provides a direct means to measure R_{Ii}.

In essence, Equation (40) defines the concept of information work, at least in the context of goal-driven information flows. It couples information to energy in a way that is consistent with Shannon's definition of information. It is also independent of the content of the information that is flowing.

Equation (40) suggests three properties of information devices: device resistance, device capacity and conversion efficiency. All of the information devices in this model create some resistance to information flow that only work can overcome. That resistance is dependent upon a device's implementation. The device resistance, R_{Ii}, characterizes the information handling properties of the device independent of the rate of information flow.

Equation (40) implies that maintaining the existence of an abstract world imposes a load upon the capacities of the information devices composing an information system. Ideally, the device resistance is constant over a useful range of information flow rates.

Despite the importance of force-driven information flows, they alone cannot explain all information flows since some occur without regard for purpose or intent. Previous arguments describing the behavior of information system complexity, as required by the Second Law of Thermodynamics, suggest the statistical underpinnings of another information flow mechanism, diffusion. Entropy or complexity differences within an information system can drive information flows. These flows behave just as most leakage and dissipation phenomena because they are all driven by differences in entropy. From the Second Law of Thermodynamics, information will tend to diffuse within an information system (i.e., flow through communications, memory and processing). Long before key developments in statistical thermodynamics provided a stochastic explanation for diffusion available, Fick's law captured the essence of diffusive behavior from empirical observations [46]. This empirical law, which applies to many different substances and media, serves as the basis for this paper's approximation of information diffusion.

A Fick's law for information diffusion could be

$$I_{Di} = D_{Si} \nabla_T \rho_{Ii} \tag{41}$$

where I_{Di} = rate of information flow due to diffusion across the boundaries of the information device or system i,

D_{Si} = diffusion constant for that information through the system, and

$\nabla_T \rho_{Ii}$ = topological difference in information density between the device or system i and the other components with which it has connectivity.

The ∇_T operator used here is not the typical spatial difference operator used commonly in traditional physics, hence the T subscript. In this case, it represents the differential operator applied in the topological space of a device. In this space, each connection between a device or system represents a different direction, here assumed orthogonal since a flow in one connections does not necessarily imply flows in any of the others. Further, the density describes the amount of information related to a specific concept. So, this application of Fick's law applies to the system's topological space and content space. It states that information will tend to diffuse from one part of a system to another until the system contains the same information content everywhere. Again, as with most of the relationships proposed herein, Equation (41) is a linear approximation of what relationships might really exist. Only experimentation and observation can determine the accuracy of this and the other approximations proposed within this paper.

The argument that this paper presents is that entropy differences, captured by the content differences, really drive these diffusive flows and they have a statistical interpretation as does diffusion in other physical systems [46, 47]. Entropy, here, is physical entropy rather than that defined by information theory although some have argued their connection [19, 47].

Diffusive flows within an information system are inevitable and only those forces imposed by system goals can retard entropy-driven information leakage. These can take the form of information security mechanisms in organizations. The existence of these forces implies that an information system must perform work to prevent information leakage. The barriers to information leakage only reduce the probability of leakage to approach but never equal zero. However, despite their pervasive influence, entropic forces are generally much smaller than other forces so the information flows they drive are much smaller. This difference permits ignoring their influence in many situations.

This discussion of information flow mechanisms suggests Assumption 17.

Assumption 17: All of the information flows within an information system obey the following relationship

$$I_{IS} = I_{IG} + I_{ID} \tag{42}$$

where I_{IS} = total information flow within an information system,

I_{IG} = information flow within the information system driven by goal forces, and

I_D = information flow occurring because of diffusion within the information system.

As with other types of flows, diffusive flows occur without regard to and, therefore, independent of any intent held by the system. Impairment of diffusive flows requires the existence of the goals and the mechanisms to specifically recognize and retard them. Force-driven flows result completely from system intent and the perceived differences between intended state and actual state. Together, these two phenomena account for all of the primary information flows within all information systems.

Information System State & Behavior

Information systems assemble the finite set of symbols contained by their information devices into abstract worlds. An information system's devices transform the energy from the physical world into the symbols representing that abstract world and that world exists only within those devices and the information devices with which they communicate. Within that world

- Objects exist,
- Objects interact with one another, and
- Objects change state over the course of time.

As Definition 2 suggests, only those symbols that contribute to an information system's abstract world representation have meaning for that system. As described above, an information system composes its symbols into concepts and combines those concepts to represent the state and behavior of the abstract world. The property values describing object state collectively depict an abstract world's state and the dependencies describing object property interactions portray an abstract world's behavior. The behavior of that abstract world completely determines the content of the information produced as output in response to the content of the information the system receives as input. Depending upon the component device capabilities, an information system's abstract world can be extremely complex and internally consistent. The abstract world created by each information system may or may not correspond closely with the abstract worlds of other information systems. Lack of correspondence between the abstract worlds of interacting information systems may be the root of many interoperability problems seen in real systems today.

An information system's memory devices maintain persistent abstract object property state and so the persistent state of the abstract world. The number of represented properties define the dimensionality of that world. Objects interact through their executable dependencies within this world and the states of these objects change over time due to these interactions. An information system's processors, executing the instructions associated with these dependencies, manifest these abstract object interactions. The possible ranges over which object property values can vary through these interactions define the accessible volume of an abstract world. Communication components can also connect an information system to relevant elements of the external world where necessary. Their extent and characteristics determine the limits to which an information system can respond to the external environment and may determine the content of the information received as input.

As with all information, each object property and dependency occupies a volume given by the total number of symbols required to uniquely specify that property state or executable dependency. An information system's execution of its executable information completely determines the changes it makes to the information content it represents and, therefore, to the state of its abstract world. Conversely, an abstract world's dependencies must be executable by the information system's devices in order to be manifested within that system.

The capabilities of any information system are limited by the capabilities of the devices that comprise it. A system's capabilities can never exceed the sum of the capabilities of the individual devices composing it. The extent of the abstract world represented by an information system's information depends upon the capacities of the devices that create that world. Device capabilities define the limits of the abstract world that an information system can support. In this sense, they create one set of boundary conditions that constrain the nature of these abstract worlds. Information systems place ultimate limits on the size of the abstract space represented through storage and encoding limits. Information devices also place limits upon the rate at which state transitions occur by their processing and communication bandwidth limitations.

As previously mentioned, information, within an information system, exhibits the property of locality. The association of particular information with specific devices and with their locations in the system topology locates information content within the system. Information can only affect a device's behavior if that information exists within that device at the time it executes a dependency. For some information to impact a device's behavior that does not possess it, that information must be transported to that device's physical location. Therefore, the set of all symbols that an information system's devices represent together with their topological location at any time completely describes the state of that system's information at that time. The set of all instructions that an information system's devices are executing and the topological location of their execution at any time completely describes the behavior of that system at that time. This location in information system space may be more important than information's physical location because the topological location provides access to connectivity of the system. An information system's device topology constrains the possible interactions within the abstract world.

State and behavior manifests first at the physical location in the system where the dependencies producing that behavior are executed and where the data exists to produce the observed state change. This locality property has two effects:

- Behavior will be manifested only where dependencies are executed; and
- Object dependencies can only be manifested between objects that have device connectivity between object locations.

The objects in the abstract world exist as entities localized within the device topology of the information system. Object state can only exist where there exists enough memory device capacity to hold those object states. Object behavior can only occur where there exists enough device memory to hold the dependency instructions and the processing capability to execute those instructions. As a result, object state dependencies can only exist if information can flow from one object state representation, through an executing dependency, to another object state representation. If the independent and dependent state representations exist in different parts of the information system topology then there must exist the connectivity between those two parts to support the required information flows. In general, these topologies are limited and these limitations significantly constrain the information flows, and therefore the behaviors, that are possible.

Equations (34) and (41) suggest a connection between complexity changes and information device latency. For processors, the latency required to output a concept from an input concept, the processing latency, is approximated by

$$L_{Eij} = (G_{Di} \mid C_{Oi} - C_{Ii} \mid) / H_{Pj} \tag{43}$$

where L_{Eij} = processing latency,

G_{Di} = dependency workload,

C_{Oi} = output data complexity,

C_{Ii} = input data complexity, and

H_{Pj} = processor work capacity.

The dependency workload is a property of the dependency being executed. The data complexity changes describe the differences between the complexity of the input and the output created by execution of the dependency. This is a property of the input and output concepts. The processor work capacity is a property of the device performing the executions. Equation (43) separates the influences of the dependency, device and input and output data streams.

For communication links, the latency from the time data enters a transmitter until the time corresponding data leaves the receiver is approximated by

$$L_{Cij} = (G_{Ei}\, C_{Ti}) / H_{Tj} + L_{Pjk} + (B_{Rk}\, V_{Ri}\, R_{Ei}) + (G_{Dk}\, C_{Ri}) / H_{Rk} \tag{44}$$

where L_{Cij} = communication link latency,

G_{Ei} = encoding workload,

C_{Ti} = transmitted data complexity,

H_{Tj} = transmitter work capacity,

L_{Pjk} = propagation latency from transmitter j to receiver k,

B_{Rk} = received bandwidth,

V_{Ri} = input data volume,

R_{Ei} = encoding compression ratio,

G_{Dk} = decoding workload,

C_{Ri} = received data complexity, and

H_{Rk} = receiver work capacity.

Transmitter and receiver work capacities are properties of the devices. Encoding and decoding loads and the encoding compression ratio are properties of the dependencies that perform the encoding and decoding operations. These dependencies are executed by the transmitter and receiver devices. When these dependencies are integral with the transmitter and receiver, the encoding and decoding loads can be associated with the devices as well. The propagation latency is a property of the communication channel between the transmitter and receiver. In many cases, the latencies associated with encoding and decoding can be ignored because those components contribute a small amount to the latency. The received bandwidth is a property of the communication link components. The data complexities are properties of the data being handled. In some cases, the decoded data is not the same as the encoded data because of the effects of lossy encoding techniques.

Two latencies are associated with information storage systems, access latency and the storage latency.

The latency from the time a query enters the information storage system to the time data is output in response, the access latency, is approximated by

$$L_{Aj} = ((G_{Qi}\, C_{Qi}) + (G_{Ci}\, C_{Oi})) / H_{Aj} + L_{Rj} + (V_{Oi} / B_{Aj}) \tag{45}$$

where L_{Aj} = memory access latency,

G_{Qi} = query handling workload,

C_{Qi} = query complexity,

G_{Ci} = decoding workload,

H_{Aj} = reader work capacity,

L_{Rj} = medium seek latency,

V_{Oi} = output data volume, and

B_{Aj} = reading bandwidth.

The query handling load is a property of the dependency that the memory reader uses to interpret queries. The query complexity and output volume are properties of the data containing the query and the output, respectively. The medium seek time and reading bandwidth are both properties of the memory reader and the storage medium.

The latency to write the contents of a data upon the storage medium is approximated by the storage latency such that

$$L_{Sj} = (G_{Ei} \, C_{Ii}) / H_{Wj} + L_{Wj} + (V_{Ii} \, R_{Ej}) / B_{Wj} \qquad (46)$$

where L_{Sj} = memory storage latency,

H_{Wj} = writer work capacity,

L_{Wj} = medium write latency,

V_{Ii} = input data volume,

R_{Ej} = encoding compression ratio, and

B_{Wj} = writing bandwidth.

The encoding load and encoding compression ratio are properties of the dependency that encodes data for storage. This characteristic can be associated with the memory writing device if it is not changed. The data complexity and volume are properties of the input data being stored. The writer work capacity is a property of the device. The medium write latency and writing bandwidth are properties of the device and the storage medium.

Summary & Conclusions

The information system model, proposed in this paper, assumes information flow and storage sufficiently describe the behavior of all information systems and then advances several relationships between the nature of information devices, information volume and information content that quantitatively describe those phenomena. This development contains no assumptions, explicit or implicit, regarding the purpose, implementation or scale of the information systems explained. This description could be applied to a room full of interacting people as well as a global information infrastructure. Given this generality, this model might appear useful without further development. In fact, it has already provided a useful framework in itself for evaluating and comparing information system architectures for various purposes [48, 49].

However, the relationships proposed here, while developed as rigorously as possible and from a sound base of assumptions, have not been tested experimentally. The relationships advanced by this theory describe its hypotheses. These hypotheses are intuitively appealing and appear to conform to empirical observations of several different types of information systems at many different levels of abstraction. However, despite their broad appeal, these hypotheses are neither complete nor have they been rigorously tested. Fortunately, all of the proposed hypotheses are described in measurable terms and, thus, can be experimentally validated.

The next step in the evolution of the physics proposed herein is to test these hypotheses through a systematic and rigorous experimental process. Experimental observations can come from many different forms of experiments. Carefully controlled experiments of traditional design represent one end of this spectrum of possibilities. In this form of experiment, an information system would be instrumented and exercised through a set of narrowly defined conditions. Each path would consider the influence of varying only one parameter if feasible. This class of experiment is likely possible only on very small systems whose states can be rigorously controlled and measured.

Observations of real information systems performing their day-to-day functions represent the other end of the experimental spectrum. In these experiments, the observer cannot control the state of the observed system and may only be able to measure some subset of its states. In a sense, the observer is completely uninvolved in the system's operation and resembles an astronomer observing the behavior of a star or galaxy. Both types of experiments require careful characterization of measurement errors. This may require many repetitions of the same experimental observations to accurately define the limits of uncontrollable fluctuations. Fortunately, traditional science has acquired a respectable body of knowledge on experimental design that can directly benefit information physics experiments. The ability to rely upon this guidance is one of the important strengths of the proposed model.

Currently, work is underway to find support for performing some set of these experiments. The results from these experiments will create the next layer of knowledge of information physics. These results may confirm or refute the relationships proposed in this paper. Most likely, these careful and systematic observations of information system behavior will cause their revision and lead to still more experiments. In the end, this path to understanding information system phenomena will significantly improve our abilities to construct and understand the behavior of complex information systems. The model discussed in this paper takes a first step in constructing a stable technical base upon which to build the systematic experimental exploration and characterization of information physics phenomena.

References

[1] R.W. Landauer, "Information Is Inevitably Physical," Feynman and Computation, A.J.G. Hey, ed., Perseus Books Publishing, Reading, MA, 1999, pp77-92.

[2] R.W. Landauer, "Information Is Physical," Physics Today, **44** (5), 1991, pp23-29.

[3] R.W. Landauer, "The Physical Nature of Information," Physics Lett. A, **217** (4-5), 1996, pp187-193.

[4] F.W. Kantor, Information Mechanics, John Wiley & Sons, New York, NY, 1977.

[5] C.E. Shannon & W. Weaver, The Mathematical Theory of Communication, Univ. of Illinois Press, Urbana, IL, 1963.

[6] R.B. Ash, Information Theory, Dover Publications, New York, NY, 1965.

[7] T.M. Cover & J.A. Thomas, Elements of Information Theory, John Wiley and Sons, New York, NY, USA, 1991.

[8] W.C. Athas, J.G. Koller & L.J. Svensson, "An Energy-Efficient CMOS Line Driver Using Adiabatic Switching," Proc. 4th IEEE Symp. on VLSI Design, 1994, IEEE Press, Piscataway, NJ, USA, pp159-164.

[9] W.C. Athas & L.J. Svensson, "Reversible Logic Issues in Adiabatic CMOS," Proc. 3rd IEEE Workshop on Physics & Computation, Dallas, TX, 1994, IEEE Computer Society Press, Los Alamitos, CA, pp111-118.

[10] W.C. Athas, L.J. Svensson, J.G. Koller, N. Tzartzanis & E.Y.-C. Chou, "Low-Power Digital Systems Based on Adiabatic-Switching Principles," IEEE Trans. on VLSI Systems, **2** (4), 1994, pp398-407.

[11] T.F. Knight, Jr., Reversible Computing for Energy Efficient and Trustable Computation, Massachusetts Institute of Technology, Cambridge, MA, 1995.

[12] M.P. Frank, T.F. Knight, Jr. & N.H. Margolus, Reversibility in Optimally Scalable Computer Architectures, Massachusetts Institute of Technology, Cambridge, MA, 1997.

[13] S.G. Younis & T.F. Knight, Jr., "Practical Implementation of Charge Recovering Asymptotycally Zero Power CMOS," Proc. 1993 Symp. on Integrated Systems, 1993, MIT Press, Cambridge, MA, USA, pp234-250.

[14] R.W. Landauer, "Irreversibility and Heat Generation in the Computing Process," IBM J. of Research & Development, **5**, 1961, pp183-191.

[15] E. Fredkin & T. Toffoli, "Conservative Logic," Int. J. of Theoretical Physics, **21** (3/4), 1982, pp219-253.

[16] C.H. Bennett & R.W. Landauer, "The Fundamental Physical Limits of Computation," Scientific American, **253** (1), 1985, pp48-56.

[17] C.H. Bennett, "Logical Reversibility of Computation," IBM J. of Research & Development, **17**, 1973, pp525-532.

[18] C.H. Bennett, "The Thermodynamics of Computation - A Review," Int. J. of Theoretical Physics, **21** (12), 1982, pp905-940.

[19] R.P. Feynman, Feynman Lectures on Computation, Addison-Wesley Publishing Co., Inc., Reading, MA, 1996.

[20] K.K. Likharev, "Classical and Quantum Limitations on Energy Consumption in Computation," Int. J. of Theoretical Physics, **21** (3/4), 1982, pp311-326.

[21] K.K. Likharev, "Rapid Single-Flux-Quantum Logic," Proc. 4th Workshop on Physics & Computation, Boston, MA, USA, 1996, New England Complex Systems Institute, Cambridge, MA, USA, pp194-195.

[22] K.K. Likharev & A.N. Korotkov, "Reversible Computation in a System with Discrete Eigenstates," Proc. 4th Workshop on Physics & Computation, Boston, MA, USA, 1996, New England Complex Systems Institute, Cambridge, MA, USA, pp194-195.

[23] M. Li & P.M.B. Vitanyi, "Theory of Thermodynamics of Computation," Proc. 2nd IEEE Workshop on Physics & Computation, Dallas, TX, 1993, IEEE Computer Society Press, Los Alamitos, CA, pp42-46.

[24] A.M. Turing, "On Computable Numbers, with an Application to the Entscheidungsproblem," Proc. London Mathematical Society, Series 2, **42**, 1936-1937, pp230-265.

[25] L. Brillouin, Science and Information Theory, Academic Press, New York, NY, 1956.

[26] C.L. Saxe, "Microprocessors," The Handbook of Computers and Computing, A.H. Seidman and I. Flores, eds., Van Nostrand Reinhold Co., New York, NY, 1984, pp62-85.

[27] M. Davis, Computability and Unsolvability, Dover Publications, Inc., New York, NY, 1982.

[28] E. Gurari, An Introduction to the Theory of Computation, Computer Science Press, Rockville, MD, 1989.

[29] C.H. Bennett, "How to Define Complexity in Physics, and Why," Complexity, Entropy and the Physics of Information, W.H. Zurek, ed., Perseus Books, Reading, MA, 1990, pp137-148.

[30] C.H. Bennett, "Logical Depth and Other Algorithmically Defined Properties of Finite Objects," Proc. 2nd IEEE Workshop on Physics & Computation, Dallas, TX, 1993, IEEE Computer Society Press, Los Alamitos, CA, pp75-77.

[31] W.H. Zurek, "Algorithmic Randomness and Physical Entropy," Physical Review A, **40** (8), 1989, pp4731-4751.

[32] W.H. Zurek, "Algorithmic Randomness, Physical Entropy, Measurements, and the Demon of Choice," Feynman and Computation: Exploring the Limits of Computers, A.J.G. Hey, ed., Perseus Books, Reading, MA, 1999, pp393-410.

[33] M. Li & P.M.B. Vitanyi, An Introduction to Kolmogorov Complexity and Its Applications, 2nd ed., Springer-Verlag, New York, NY, 1997.

[34] D.C. Gross et al., "Report from the Fidelity Implementation Study Group," Paper No. 99S-SIW-167, Proc. of 1999 Spring Simulation Interoperability Workshop, Orlando, FL, March 1999, np.

[35] S. Barta, "Relation between Information and Thermodynamic Entropy," J. of Electrical Engineering, **48** (7-8), 1997, pp169-174.

[36] J.D. Bekenstein, "Entropy Content and Information Flow in Systems with Limited Energy," Physical Review D, **30**, 1984, pp1669-1679.

[37] L. Brillouin, "Maxwell's Demon Cannot Operate: Information and Entropy," Int. J. of Applied Physics, **22**, 1950, pp334-337.

[38] P.A. Dufort & C.J. Lumsden, "The Complexity and Entropy of Turing Machines," Proc. 3rd IEEE Workshop on Physics & Computation, Dallas, TX, 1994, IEEE Computer Society Press, Los Alamitos, CA, pp227-232.

[39] N. Fuchikami, H. Iwata & S. Ishioka, "Thermodynamic Entropy of Computer Devices," J. Physical Soc. of Japan, **68** (12), 1999, pp3751-3754.

[40] H.S. Leff & A.F. Rex, Maxwell's Demon: Entropy, Information, Computing, Princeton University Press, Princeton, NJ, 1990.

[41] J. Machta, "Entropy, Information, and Computation," American J. of Physics, **67** (12), 1999, pp1074-1077.

[42] P.L. Patterson, "Entropy, Fault Tolerance, and Multicomputer Networks," Proc. 2nd IEEE Workshop on Physics & Computation, Dallas, TX, 1993, IEEE Computer Society Press, Los Alamitos, CA, pp232-236.

[43] T. Stonier, "Towards a General Theory of Information: Information and Entropy," Future Computing Systems, **2** (4), 1990, pp409-427.

[44] W. Yourgrau, A. van der Merwe & G. Raw, Treatise on Irreversible and Statistical Thermophysics, The Macmillan Co., New York, NY, 1966.

[45] R.T. Weidner & R.L. Sells, Elementary Classical Physics, Allyn and Bacon, Inc., Boston, MA, 1966.

[46] J. Keizer, Statistical Thermodynamics of Nonequilibrium Processes, Springer-Verlag, New York, NY, 1987.

[47] A.J.G. Hey, ed., Feynman and Computation, Perseus Books Publishing, Reading, MA, 1999.

[48] S.Y. Harmon, "Evaluating and Comparing Information Systems," Proc. of the 1998 IEEE International Conf. on Systems, Man, and Cybernetics, San Diego, CA, October 1998, np.

[49] S.Y. Harmon, "Evaluating the Performance of Intelligent Systems," Proc. of the 3rd World Multiconference on Systemics, Cybernetics and Informatics, Orlando, FL, July 1999, np.

Acknowledgements

The research documented in this paper would not have been possible without the support of the Air Force Research Laboratories and the Defense Advanced Research Projects Agency and its Information Systems Office. The author greatly appreciates the support of Mr. Larry Willis, Manager of the Wargaming for the Asymmetric Environment Program, Mr. Michael Skroch, Manager of the Information Assurance Science and Engineering Technology Program, and Ms. Deborah Cerino of the Air Force Research Laboratories.

Appendix: Table of Symbols Used in Equations

Table of Symbols Used in Equations

Symbol	Definition	Defining Equation
ΔC_{Di}	complexity change made to the system's information in performing that execution	33
$\nabla_T \rho_{Ii}$	topological difference in information density between the device or system i and the other components with which it has connectivity	41
η_{Di}	conversion efficiency for information device i	19
η_{Sij}	efficiency with which device j executes symbol j	23
Δt	time interval over which the information flow is sustained	39
a_{Ii}	constant relating goal force to information flow rate	37
a_J	constant that relates data flow to work	34
a_W	constant that relates the units of complexity change to the units of execution work	34
B_{Aj}	reading bandwidth	45
B_{Rk}	received bandwidth	44
B_{Wj}	writing bandwidth	46
C_{Ii}	input data complexity	43
C_{Oi}	output data complexity	43
C_{Oi}	output data complexity	45
C_{Qi}	query complexity	45
C_{Ri}	received data complexity	44
C_S	total complexity of an information system	27
C_{SD}	complexity contributed by the devices and media the system contains	27
C_{SI}	complexity of an information system created by the existence of meaningful information within that system	26

Table of Symbols Used in Equations (continued)

Symbol	Definition	Defining Equation
C_{SIi}	total number of dependencies whose execution could take the system to state i from any other state	28
C_{Ti}	transmitted data complexity	44
D_{Si}	diffusion constant for the path i through the system	41
E_G	force manifested by information system goals	35
G_{Ci}	decoding workload	45
G_{Di}	dependency workload	43
G_{Dk}	decoding workload	44
G_{Ei}	encoding workload	44
G_{Qi}	query handling workload	45
H_{Aj}	reader work capacity	45
H_{Pj}	processor work capacity	43
H_{Rk}	receiver work capacity	44
H_{Tj}	transmitter work capacity	44
H_{Wj}	writer work capacity	46
I_{Ci}	information flow created by device i	21
I_D	information flow occurring because of diffusion within the information system	42
I_{Di}	rate of information flow due to diffusion across the boundaries of the information device or system i	41
I_{Ei}	information flow from information erasure within device i	21
I_{IG}	information flow within the information system driven by goal forces	42
I_{IGi}	total information flow through device i caused by the existence of the information system's goals	35
I_{Ii}	information flow into the device i	21
I_{IS}	total information flow within an information system	42

Table of Symbols Used in Equations (continued).

Symbol	Definition	Defining Equation
I_{Oi}	information flow out of the device i,	
I_{Ri}	information flow retrieved from local storage within device i	21
I_{Si}	information flow stored to the local information storage within device i	21
J_{IEi}	energy flow dissipated only because of the erasure of information	4
J_{IIi}	energy flow into the device i through the flow of input information	1
J_{IOi}	energy flow out of the device i through the output information	1
J_{ISi}	energy flow into and out of the device i's local information store	1
J_{N0i}	energy flow the information device i dissipates independent of the information flow through the device	15
J_{NDi}	energy flow dissipated only because of the inefficiencies inherent to the device i	4
J_{NIi}	energy flow into the device i that does not contain information meaningful to the device	1
J_{NJi}	energy flow the information device i dissipates as a function of the information flow through the device	15
J_{NOi}	energy flow out of the device i that does not contain information meaningful to the device	1
k	Boltzmann's constant	29
L_{Aj}	memory access latency	45
L_{Cij}	communication link latency	44
L_{Eij}	processing latency	43
L_{Pjk}	propagation latency from transmitter j to receiver k	44
L_{Rj}	medium seek latency	45

L_{Sj}	memory storage latency	46

Table of Symbols Used in Equations (continued).

Symbol	Definition	Defining Equation
L_{Wj}	medium write latency	46
N_{Dij}	number of object properties that specify the dependent variables of the jth dependency of object instance i	26
N_{DPi}	number of standard instructions executed per unit volume of input data	25
N_{Ii}	number of standard instructions executed per dependency	24
N_{Pij}	number of object properties that specify the independent variables of the jth dependency of object instance i	26
P_{0i}	power the device i dissipates when all information flows througy it are zero	17
p_i	probability of reaching system state i	29
p_{Ii}	probability that a system's information state will be in state i	28
P_{Ii}	power the device i dissipates only as the result of information flowing through it	17
P_{Oi}	total power dissipated by device i over time	17
R_{Ei}	encoding compression ratio	44
R_{Ii}	device resistance to information flow	37
S	system entropy	29
V_D	data flow volume required to produce the output	34
V_{Ii}	input data volume	46
V_{Ii}	volume of input data	25
V_{Oi}	output data volume	45
V_{Ri}	input data volume	44
W_{Di}	work a system performs in executing a dependency i	24
W_{Ii}	work performed by the device i	40

W_{ij}	work device i invests to execute symbol j	23

Table of Symbols Used in Equations (continued).

Symbol	Definition	Defining Equation
w_j	normalized work required for any device in an information system to execute symbol j	23
W_O	work required to produce an output	34

*U.S. GOVERNMENT PRINTING OFFICE: 2001-610-055-10089

MISSION
OF
AFRL/INFORMATION DIRECTORATE (IF)

The advancement and application of Information Systems Science and Technology to meet Air Force unique requirements for Information Dominance and its transition to aerospace systems to meet Air Force needs.